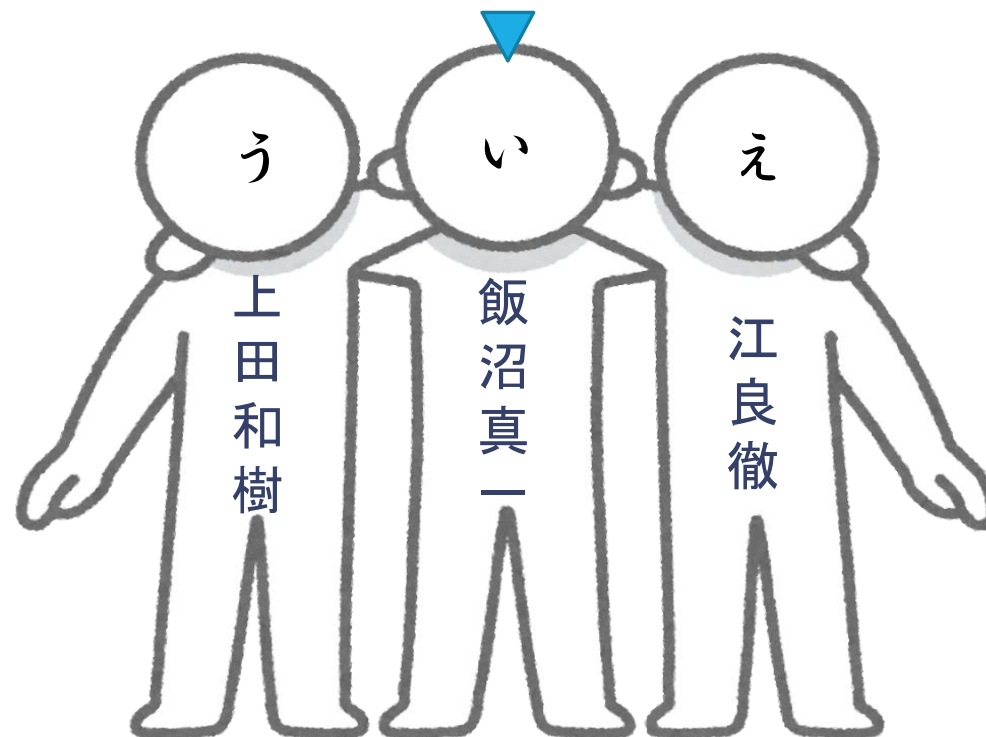


サクセス(高品質)ストーリー

～加速スイングバイを使え～

チーム名：出席番号となり同士



1. 販売戦略
2. テスト戦略
3. テスト全体計画
4. テスト要求分析
5. テスト詳細設計
6. まとめ

1.販売戦略

販売戦略

テスト戦略

全体計画

要求分析

詳細設計

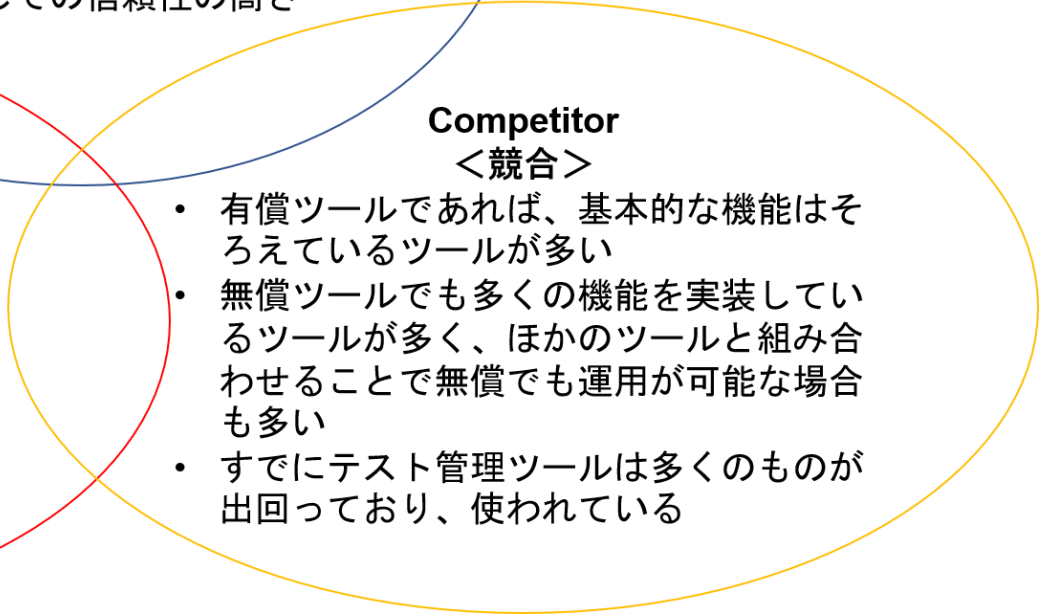
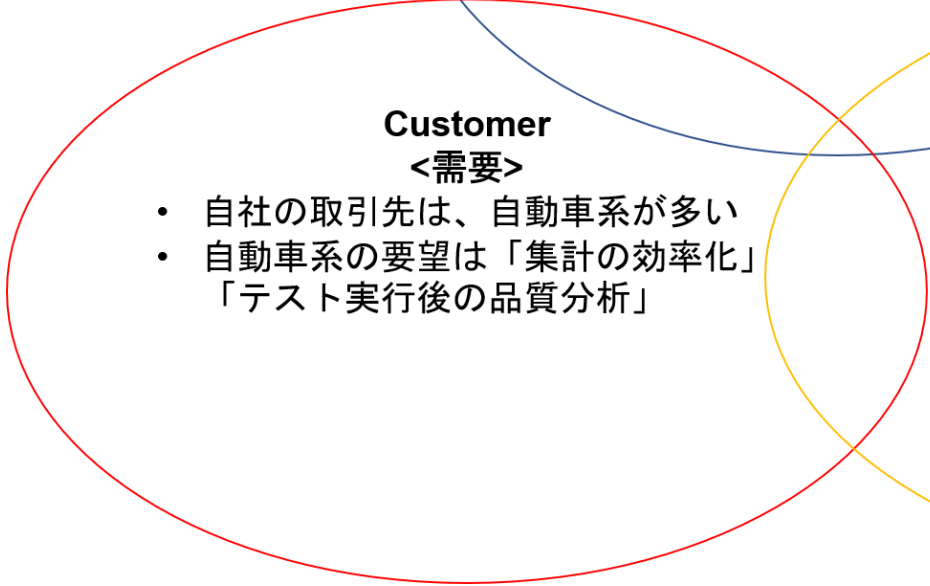
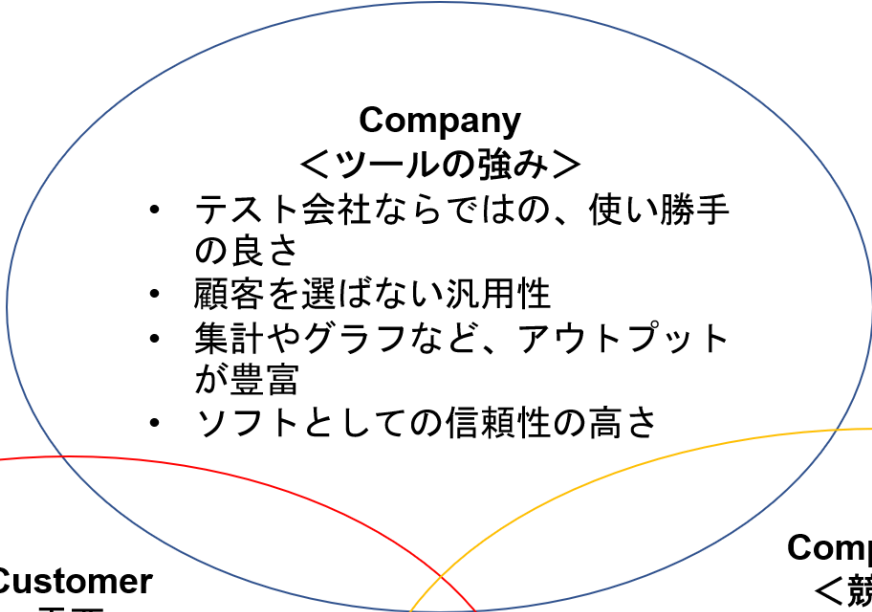
まとめ

Quality Forwardを販売したい

どのような戦略で販売するか

マーケット戦略部門からの報告書が上がってきた

3C分析の結果



- Company**
<ツールの強み>
- テスト会社ならではの、使い勝手の良さ
 - 顧客を選ばない汎用性
 - 集計やグラフなど、アウトプットが豊富
 - ソフトとしての信頼性の高さ

- Customer**
<需要>
- 自社の取引先は、自動車系が多い
 - 自動車系の要望は「集計の効率化」「テスト実行後の品質分析」

- Competitor**
<競合>
- 有償ツールであれば、基本的な機能はそろえているツールが多い
 - 無償ツールでも多くの機能を実装しているツールが多く、ほかのツールと組み合わせることで無償でも運用が可能な場合も多い
 - すでにテスト管理ツールは多くのものが出回っており、使われている

①Company分析 ～ASTER社の取引先～

販売戦略

テスト戦略

全体計画

要求分析

詳細設計

まとめ

自動車分野の既存取引が多い

分野	売上構成比	前年比	コメント
自動車分野	54.0%	14.4	OEM、Tier1、Tier2の開拓により拡大
ITシステム分野	21.0%	40.3	Sierとの直接取引拡大
デジタル機器分野	9.4%	7.4	IOT機器が拡大
産業機器、その他分野	15.6%	39.7	医療、エンタメが拡大

②Customer分析 ～業界規模の調査～

6

販売戦略

テスト戦略

全体計画

要求分析

詳細設計

まとめ

自動車OEMの業界規模は巨大

業種	事業内容	開発製品	業界(※1)	業界規模	伸び率	コメント
OEM	自動車メーカー	自動車	自動車	69.6兆	+1.6%	市場規模は巨大
ゲーム	家庭用ゲーム機メーカー	ゲーム機本体	ゲーム	5.5兆	+12.7%	市場規模は大きくないが、伸び率は高い
車載	車載メーカー	カーナビ	自動車部品	36.9兆	+3.0%	市場規模は大きい
MAP	地図ナビアプリ開発会社	MAPアプリ	IT	15.6兆(※2)	+2.9%	MAPアプリに限定するなら、規模は小さい
メーカー	コンシューマ向け電子機器メーカー	電子辞書	家電	48.6兆(※3)	-0.8%	電子辞書に限定するなら、規模は小さい
クラウドアプリ	法人向け会計クラウドサービス	会計クラウドアプリ	IT	15.6兆(※4)	+2.9%	規模はそこそこ大きい。成長率は低い

※1 <https://gyokai-search.com/5-kibo.html>より

※2 IT業界に属するが、規模は小さいと予測

※3 家電業界に属するが、規模は小さいと予測

※4 IT業界のすべてではないが、クラウドの占める比率は相当高くなっていると予測

②Customer分析 ～各業種からの要望～

販売戦略

テスト戦略

全体計画

要求分析

詳細設計

まとめ

自動車OEM業界の要望は明確

業種	事業内容	開発製品	要望
OEM	自動車メーカー	自動車	集計業務のムダ削減(Tier1 とのリレーション)。テスト実行後の品質分析力強化
ゲーム	家庭用ゲーム機メーカー	ゲーム機本体	大量のテストマネジメントを少人数で対応する体制。 テスト項目の網羅性の分析による品質向上
車載	車載メーカー	カーナビ	大量のテスト実施状況のリアルタイム可視化。多拠点の情報一元化
MAP	地図ナビアプリ開発会社	MAPアプリ	OEM、仕向地ごとのテスト実施状況の一元化
メーカー	コンシューマ向け電子機器メーカー	電子辞書	膨大なテスト評価管理の効率化(約 6,000 枚～の評価シート)
クラウドアプリ	法人向け会計クラウドサービス	会計クラウドアプリ	リグレッションテスト業務。効率化テスト項目内容の精度向上

販売戦略

テスト戦略

全体計画

要求分析

詳細設計

まとめ

- 自動車メーカーのTier1、Tier2向けに販売
 - ▶ Tier2向けには、Tier1への報告や連携を想定
- 販売のために「集計の効率化」「テスト実行後の品質分析」の強化
- 差別化のため、パッケージ化に伴い、「セキュリティ」「可用性の高さ」の強化
- 市場シェア拡大のため、必要な個所からリリースし早期投入
- シェア拡大は顧客の囲い込みになり、「安定した保守料の獲得」「ソリューションビジネスへの誘導」へつながり、経営の安定に寄与
 - ▶ 顧客プロジェクトデータをクラウドに保持するため

販売戦略

テスト戦略

全体計画

要求分析

詳細設計

まとめ

開発チームからの方針報告(サマリ)

・「売れる」ソフトを作るために以下を方針とする

- 以下に関連する機能を優先して開発する
 - 「データ集計の効率化」
 - 「テスト実行後の品質分析」
- できるだけ短納期で開発する必要がある
 - 可能ならば、段階的にリリースしたい
 - できれば仕様書の作成は最小限にしたい
- 非機能は、以下を重点的に考慮して設計する
 - 「セキュリティ」
 - 「可用性の高さ」

2. テスト戦略



売れるシステムをリリースする

- 一度顧客を囲い込めば、安定した収入が得られる
- ツール売りだけでなく、ソリューションビジネスにつなげることができる

マーケット戦略部の販売戦略からゴール分割

- 後発のため、早期リリースが必要
- テスト管理ツールは多数リリースされているため、差別化しないと販売が難しい

早期リリースにより、管理ツール未導入ユーザーの獲得を目指す

他製品との差別化を図るため、「セキュリティ」「可用性の高さ」を非機能の重点テストとする

対象ユーザーを絞って、対象ユーザーが必要とする機能を優先開発し、不具合を抽出するピンポイントテストで品質を高める

以下がテストされている結果

- 「セキュリティ」
- 「可用性の高さ」

以下の機能についてテストされている結果

- 「集計の効率化」
- 「テスト実行後の品質分析」

少ない工数で効率の良いテストを行う

- 早期リリースといっても、品質が低いものをリリースしてしまうと後々のビジネスに影響する
- 今のテストチームでは、セキュリティテストなどの非機能テストのノウハウが少ない

より少ない工数で多くの網羅できるテストを行う

機能網羅と状態遷移と画面遷移を包含したシナリオテスト結果 (機能/画面/状態遷移の網羅率は最低でも80%以上)

粒度の粗い仕様書をテストチームにて補完する

SQuaREや利用時分析をインプットとしたテストタイプ一覧

外注の活用などで、セキュリティテストのアウトソーシングを行う

外注へのRFPと受入テスト結果

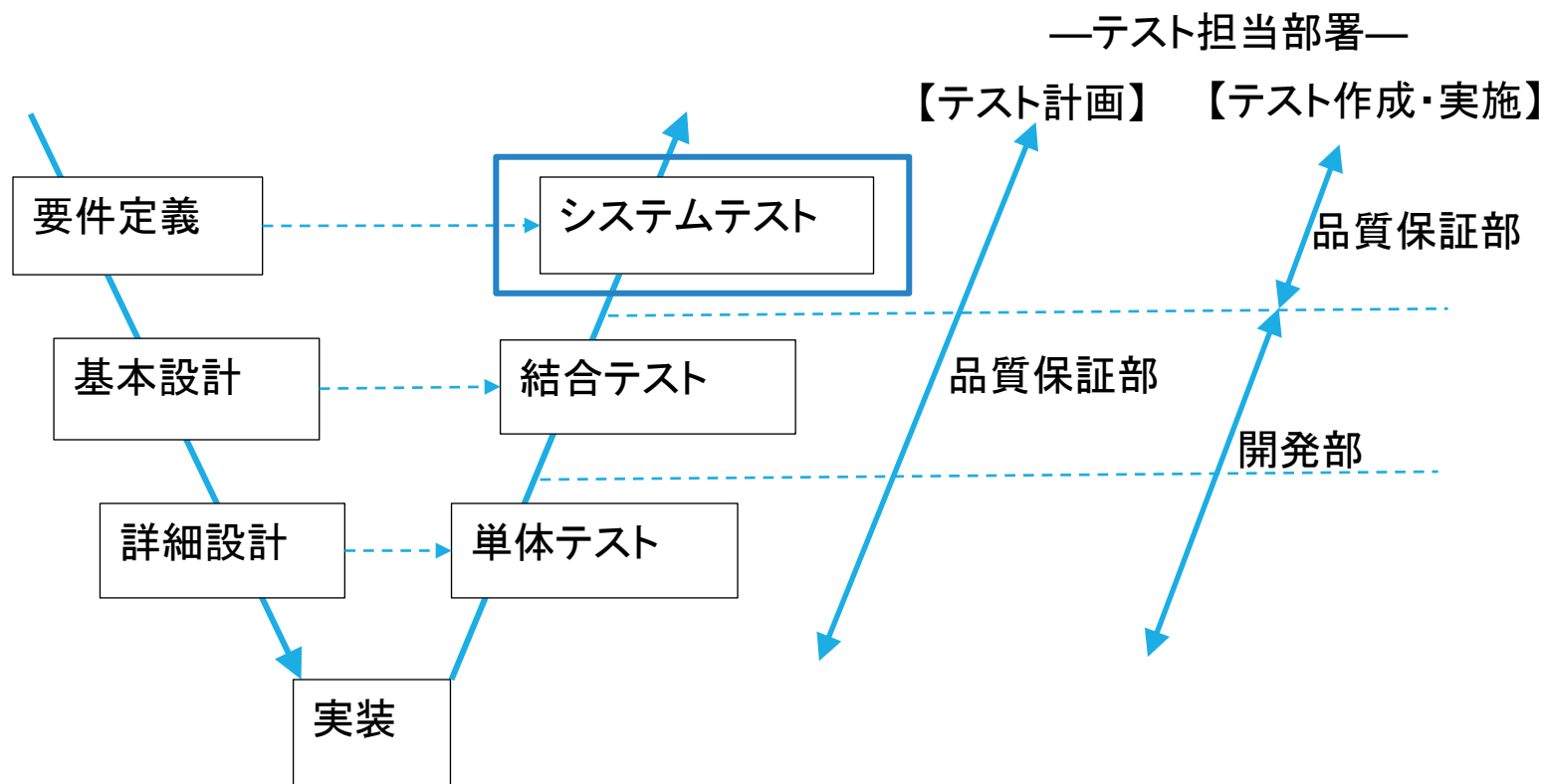
オフショアの活用によるリソースの補完

オフショアへのRFP、教育結果、受入テスト結果

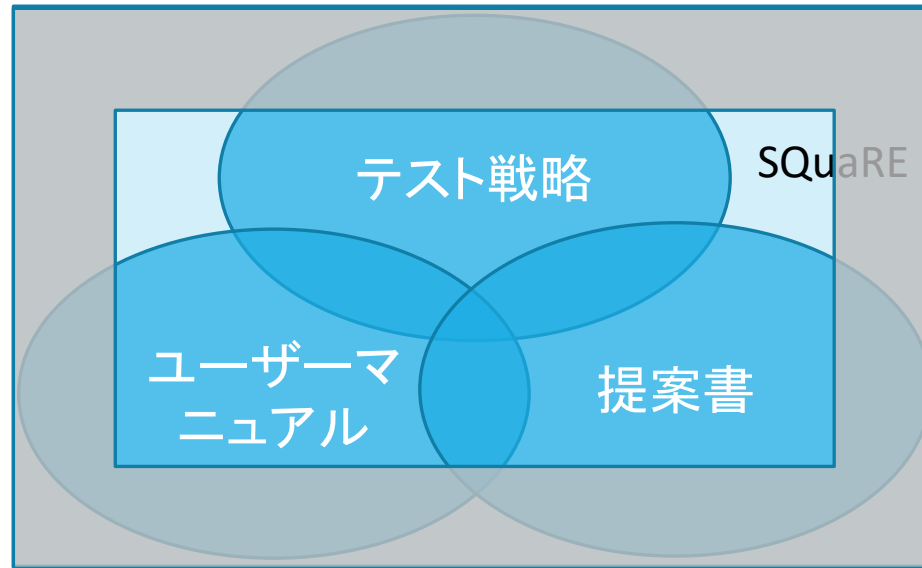
3. テスト全体計画



・ASTER社開発プロセス



4.テスト要求分析(テストタイプの特定)



- 1. 3つのインプットからテストタイプを洗い出す
- 2. SQuaREで不足しているテストタイプを補完する →15種
- 3. リスクを分析し、リスク「高」のものに絞り込む →8種

販売戦略

テスト戦略

全体計画

要求分析

詳細設計

まとめ

効率化のため、複数のテストタイプをまとめて**4つに集約**

#	テストタイプ	テスト内容	補足
1	シナリオテスト	ユーザーのテスト業務をシナリオに定義し、それに沿って動作させる。ユーザーの期待を満たしているか、ふるまいを確認する。	機能テスト、UIテストの観点を包含する。
2	セキュリティテスト (ペネトレーション)	・クラウドにデプロイした状態で、悪意ある第三者から不正なアクセスされる、脆弱性が無いかテストする	アクセス権テスト、暗号化・改ざん防止テストを包含する。 セキュリティベンダーにテストを委託する
3	ロングランテスト	・長時間運用に耐えうるシステムであることを確認する。	シナリオを、テストスクリプトで自動化し実行することで、長時間運用をシミュレートする。
4	探索的テスト	重要なポイント、障害が出そうな箇所をまとめたテストチャーターを事前に準備する。	

テスト要求モデル

販売戦略

テスト戦略

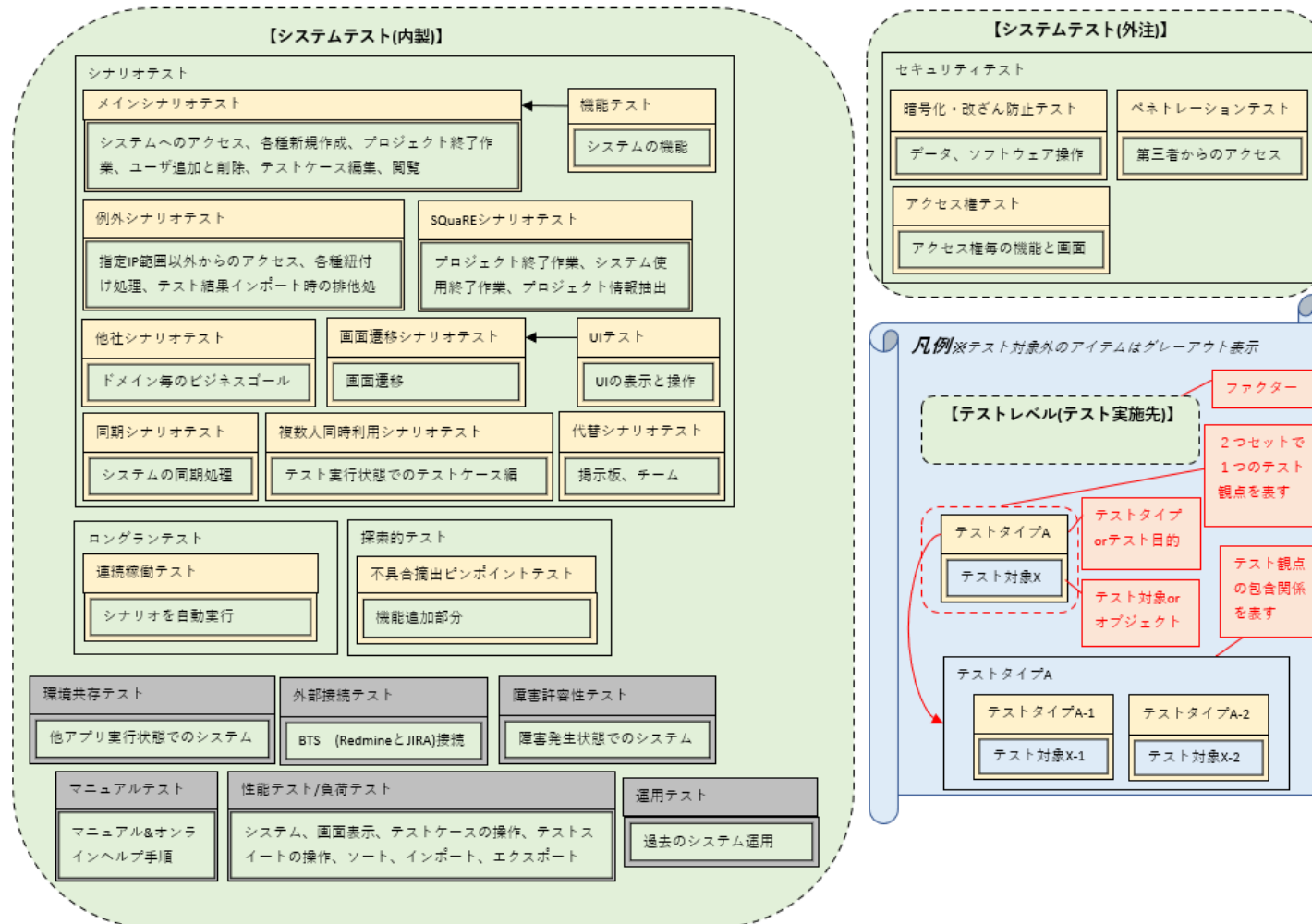
全体計画

要求分析

詳細設計

まとめ

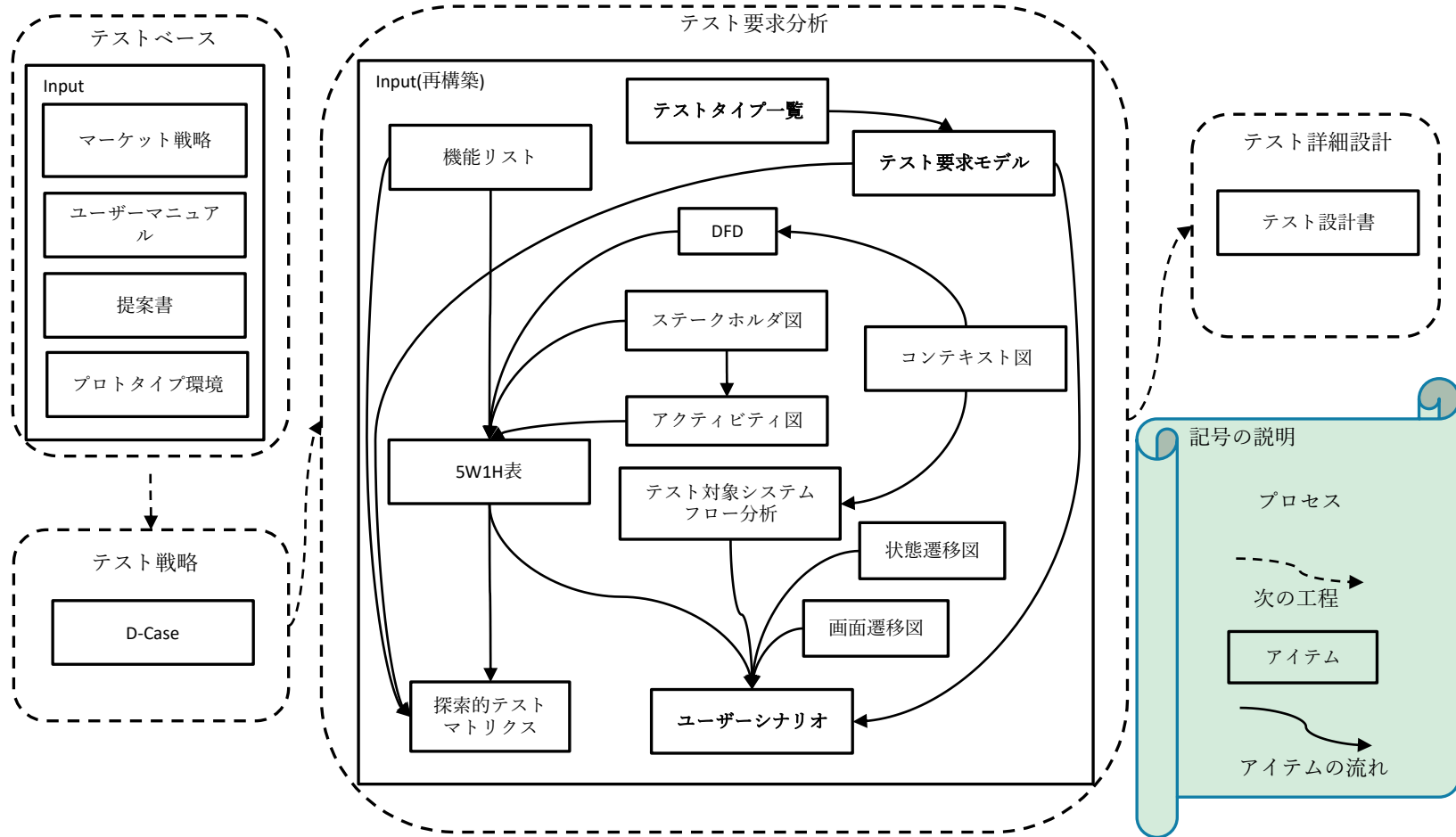
【テスト要求モデル(モデル：Processed Item Flow Diagram)】



5. テスト詳細設計 (テスト設計プロセス)



テスト設計プロセス



画面遷移図

販売戦略

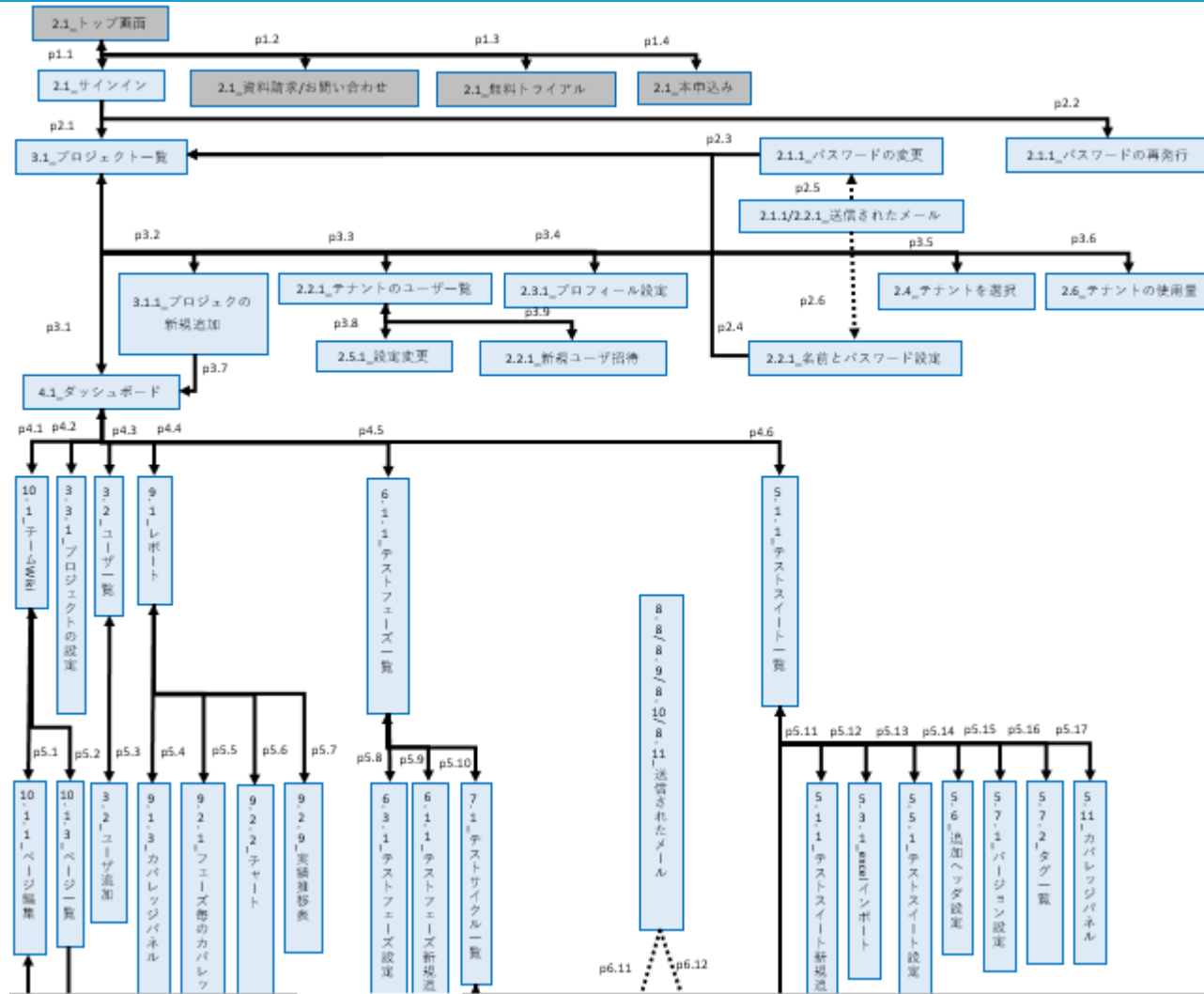
テスト戦略

全体計画

要求分析

詳細設計

まとめ



状態遷移図

販売戦略

テスト戦略

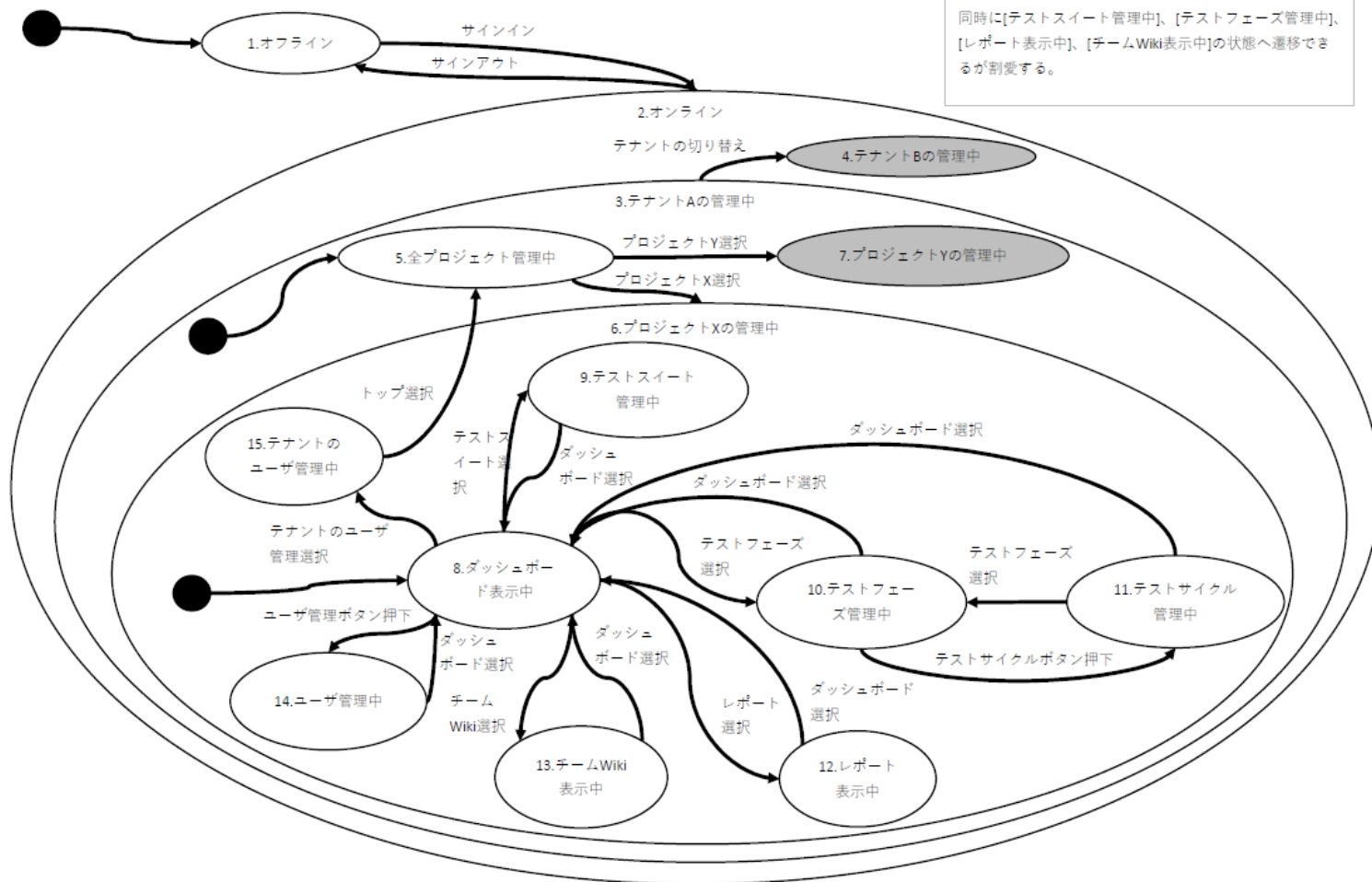
全体計画

要求分析

詳細設計

まとめ

【状態遷移図(システム全体のふるまい)】※簡易図





メインシナリオ：ソフトウェアライフサイクルのメインフローが中心		
No	メインアクター	シナリオ(ユーザのゴール)
UC01	導入時テストメンバ	テストメンバがテストスイートを作成する。添付した関連ファイルをダウンロードする。
UC02	導入時テストマネージャ	テストマネージャはテストスイートを作成してからテストフェーズを作成する。その後、カバレッジパネルに必要な項目がすべて表示されることを確認し、テストサイクルを作成する。予実設定を入力し、自動保存してから初期化する。チームWikiに書き込みする。
UC03	導入時インフラ管理者	インフラ管理者はプロジェクトを新規追加して必要なユーザを追加またはQuality Forwardへ招待する。
UC04	導入時開発者	開発者がテストケースを閲覧する。ソートし、不要な列を非表示にし、テスト結果補足の設定内容を表示する。備考を表示する。キーワード検索する。テスト
UC05	開発時テストメンバ	Excelにテストケ 完了後、チーム
UC06	開発時テストマネージャ	テストマネージャ 確認する。ラベ
UC07	開発時インフラ管理者	インフラ管理者は人員入れ替えのため、新たなテストメンバを追加してテナントのユーザ権限を変更する。その後、抜けたテストメンバを削除する。最後にテストスイート、テストフェーズ、テストサイクルに問題ないことを確認する。
UC08	開発時開発者	開発者はテスト期間中に、問題が発生しているテストケースを確認する。確認結果についてコメントを投稿する。ダッシュボード、テストスイート、テストフェーズ、テストサイクル、チームWiki、ユーザ管理からレポートを表示する。
UC09	保守時テストメンバ	テストメンバは編集箇所を検索してテストケースを編集し、フィルターする。不要テストスイートを削除とヘッダをヘルプ確認して追加してレビューする。新しいバージョンのテストスイートを作成してテスト実行する。
UC10	保守時テストマネージャ	テストマネージャは進捗管理を行い、成分バーと期日バーを確認する。テスト結果、件数と予定件数を表示する。実施中のテストフェーズ、最近完了したテストフェーズ、実施中のテストサイクル、レビュー待ちのテストサイクルを表示する。テスト結果を整理する。
UC11	保守時インフラ管理者	インフラ管理者はパスワードを再設定する。パスワードの強度を確認して変更する。プロフィール設定を変更する。テナントを切り替え、テナント使用量を確認する。終了したプロジェクト設定を変更する。APIキーを発行し、アーカイブと不要になったプロジェクトをアーカイブから戻し、プロジェクト設定を削除する。
UC12	保守時開発者	開発者はテスト結果を確認後、不要になったチームWikiのページを削除する。

12個のメインフロー
をテスト対象とする

販売戦略

テスト戦略

全体計画

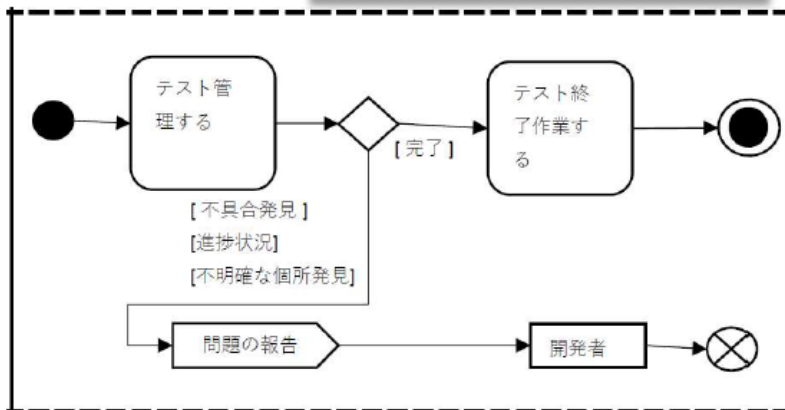
要求分析

詳細設計

まとめ

【アクティビティ図 1-6】

シナリオNo: UC6



<<前提条件>>

ソフトウェアテストライフサイクル：開発時に利用

操作ユーザ：テストマネージャ

アクセス権限：プロジェクト管理者

端末：スマホ、OS(Android)、ブラウザ(Chrome)

BTS連携：JIRA

テストタイプ：メインシナリオテスト 管理 No：UC06 優先度：B

テスト目的：テストマネージャが開発中にテストメンバからのコメントを確認後、取束曲線(変動)を全体&個別で確認する。カバレッジパネルで全体&個別を確認する。ラベル設定を行い、バグ情報をアップロードする。最後に実績推移表をダウンロードする。

- テスト手順：1. テストマネージャが QF へサインインする。
 2. 担当プロジェクトを選択してダッシュボードを表示する。
 3. テストサイクルのコメントを表示する。
 4. レポート画面を開き、プロジェクト全体のレポートを表示してから、変動タブを表示する。
 5. プロジェクト全体のカバレッジパネルを表示する。
 6. テストフェーズ毎のカバレッジパネルを表示する。
 7. テストフェーズ毎のチャートを表示してから変動タブを表示する。その後、ラベル設定を行い、バグ情報のアップロードを実行する。
 8. テストフェーズ毎の実績推移表を表示し、実績推移表をダウンロードする。
 9. レポート一覧に戻り、ダッシュボードを表示する。
 10. サインアウトする。

期待結果：コメントが表示できる。取束曲線(変動)が全体&個別で表示できる。カバレッジパネルが全体&個別で表示できる。ラベル設定ができる。バグ情報のアップロードができる。実績推移表をダウンロードできる。

備考：

販売戦略

テスト戦略

全体計画

要求分析

詳細設計

まとめ

カバレッジ目標の80%以上を達成

・画面(C0)

全画面数	54画面
テストした画面数	49画面
カバレッジ	90.74%

・機能

全機能数	255機能
テストした機能数	207機能
カバレッジ	81.18%

・画面(C1)

全画面遷移パス数	110画面
テストした画面遷移パス数	90画面
カバレッジ	81.82%

・状態遷移

全状態遷移パス数	50機能
テストした状態遷移パス数	46機能
カバレッジ	92.00%



販売戦略を基にテスト戦略を立案

テスト戦略/マニュアル/提案書/SQuaREから、網羅的にテストタイプを抽出

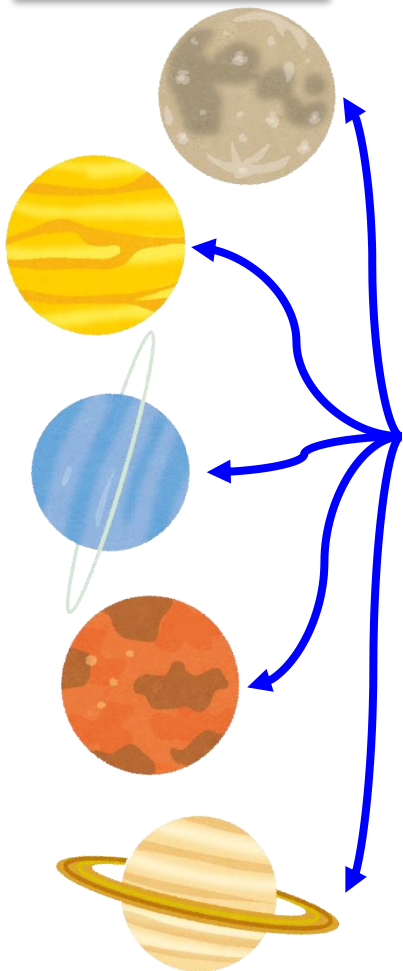
抽出したテストタイプをリスク分析し
リスク高であるテストタイプに絞り込む
「シナリオテスト」「セキュリティテスト」「ロングランテスト」「探索的テスト」(※)

様々な視点から分析を行い、それを基に「シナリオテスト」を作成

機能、画面、状態遷移のカバレッジを80%以上確保

(※)セキュリティテストは外部ベンダーに委託
ロングランテストはシナリオテストをベースに自動テスト
探索的テストはテストチャータを準備

ゴール



ループ

自動化

シナリオテスト実行

シナリオ化



不具合情報

探索的テスト



加速スイングバイ

